

CLAIMS

1. Apparatus for use with a computer system having a memory, a local storage and an existing directory service operating in the memory, the apparatus providing naming and life cycle services for a distributed object and comprising:
- a moniker object which contains an identifier that universally identifies an instance of the distributed object and a moniker name; and
 - a first stream object which automatically substitutes the moniker object for the distributed object when the distributed object is streamed out from the memory to the local storage.
2. Apparatus according to claim 1 wherein the first stream object substitutes the moniker object for the distributed object when the distributed object is persisted.
3. Apparatus according to claim 1 further comprising a second stream object which automatically substitutes a reference to the distributed object for the moniker object when the moniker object is streamed in from the local storage to the memory.
4. Apparatus according to claim 3 wherein the second stream object substitutes the moniker object for the distributed object when the distributed object is resurrected.
5. Apparatus according to claim 1 wherein life cycle services are provided by associating with the moniker object a predefined policy which specifies how and when life cycle services are performed.
6. Apparatus according to claim 5 further comprising a life cycle services object which responds to the predefined policy by controlling the life cycle of the distributed object.

- 1 7. Apparatus according to claim 1 further comprising a runtime repository which
2 includes a database of moniker name-object reference pairs.
- 1 8. Apparatus according to claim 7 further comprising a directory service factory
2 object which responds to the moniker name by instantiating a directory service
3 adapter object for applying the moniker name to the existing directory service
4 when the runtime repository does not contain the moniker name.
- 1 9. Apparatus according to claim 1 wherein the distributed object is instantiated in
2 accordance with an object model and the apparatus comprises an object model
3 adapter which processes distributed objects instantiated with the object model.
- 1 10. Apparatus according to claim 9 wherein the object model adapter returns a
2 reference to the distributed object together with a moniker object associated with
3 the distributed object.
- 1 11. A method for use with a computer system having a memory, a local storage and
2 an existing directory service operating in the memory, the method providing
3 naming and life cycle services for a distributed object and comprising the steps
4 of:
5 (a) instantiating a moniker object which contains an identifier that universally
6 identifies an instance of the distributed object and a moniker name; and
7 (b) using a first stream object to automatically substitute the moniker object
8 for the distributed object when the distributed object is streamed out from
9 the memory to the local storage.
- 1 12. A method according to claim 11 wherein step (b) comprises the step of:

- 2 (b1) using the first stream object to substitute the moniker object for the
3 distributed object when the distributed object is persisted.
- 1 13. A method according to claim 11 further comprising the step of:
2 (c) using a second stream object to automatically substitute a reference to
3 the distributed object for the moniker object when the moniker object is
4 streamed in from the local storage to the memory.
- 1 14. A method according to claim 13 wherein step (c) comprises the step of:
2 (c1) using the second stream object to substitute the moniker object for the
3 distributed object when the distributed object is resurrected.
- 1 15. A method according to claim 11 further comprising the step of:
2 (d) associating with the moniker object a predefined policy which specifies
3 how and when life cycle services are performed.
- 1 16. A method according to claim 15 further comprising the step of:
2 (e) instantiating a life cycle services object which responds to the predefined
3 policy by controlling the life cycle of the distributed object.
- 1 17. A method according to claim 11 further comprising the step of:
2 (f) creating a runtime repository which includes a database of moniker name-
3 object reference pairs.
- 1 18. A method according to claim 17 further comprising the step of:
2 (g) instantiating a directory service factory object which responds to the
3 moniker name by instantiating a directory service adapter object for
4 applying the moniker name to the existing directory service when the
5 runtime repository does not contain the moniker name.

1 19. A method according to claim 11 wherein the distributed object is instantiated in
2 accordance with an object model and wherein the method comprises the step of:
3 (h) instantiating an object model adapter which processes distributed objects
4 instantiated with the object model.

1 20. A method according to claim 19 wherein step (h) comprises the step of:
2 (h1) returning a reference to the distributed object together with a moniker
3 object associated with the distributed object.

1 21. A computer program product for use with a computer system having a memory,
2 a local storage and an existing directory service operating in the memory, the
3 computer program product providing naming and life cycle services for a
4 distributed object and comprising a computer usable medium having computer
5 readable program code thereon including:
6 class code for instantiating a moniker object which contains an identifier
7 that universally identifies an instance of the distributed object and a moniker
8 name; and
9 class code for instantiating a first stream object which automatically
10 substitutes the moniker object for the distributed object when the distributed
11 object is streamed out from the memory to the local storage.

1 22. A computer program product according to claim 21 wherein the class code for
2 instantiating a first stream object comprises method code for substituting the
3 moniker object for the distributed object when the distributed object is persisted.

1 23. A computer program product according to claim 21 further comprising class code
2 for instantiating a second stream object which automatically substitutes a

3 reference to the distributed object for the moniker object when the moniker object
4 is streamed in from the local storage to the memory.

1 24. A computer program product according to claim 23 wherein the class code for
2 instantiating the second stream object includes method code for substituting the
3 moniker object for the distributed object when the distributed object is
4 resurrected.

1 25. A computer program product according to claim 21 wherein the class code for
2 instantiating the moniker object further comprises a method for associating with
3 the moniker object a predefined policy which specifies how and when life cycle
4 services are performed.

1 26. A computer program product according to claim 25 further comprising class code
2 for instantiating a life cycle services object which responds to the predefined
3 policy by controlling the life cycle of the distributed object.

1 27. A computer program product according to claim 21 further comprising program
2 code for creating a runtime repository which includes a database of moniker
3 name-object reference pairs.

1 28. A computer program product according to claim 27 further comprising class code
2 for instantiating a directory service factory object which responds to the moniker
3 name by instantiating a directory service adapter object for applying the moniker
4 name to the existing directory service when the runtime repository does not
5 contain the moniker name.

1 29. A computer program product according to claim 21 wherein the distributed object
2 is instantiated in accordance with an object model and wherein the computer

3 program product comprises class code for instantiating an object model adapter
 4 which processes distributed objects instantiated with the object model.

1 30. A computer program product according to claim 29 wherein an instantiated
 2 object model adapter comprises a method for returning a reference to the
 3 distributed object together with a moniker object associated with the distributed
 4 object.